

Архитектура III

Кристиян Стоименов

15 ноември 2023 г.

ТУЕС,
ПВМКС



Периферии

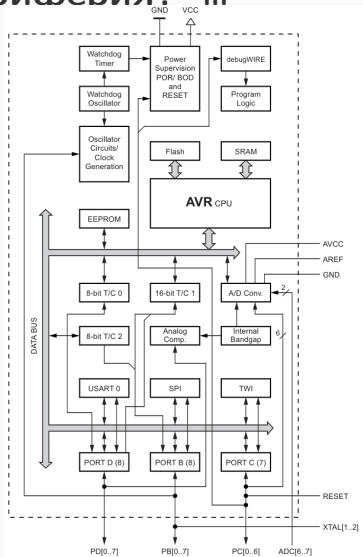
Какво е периферия? i

- Подсистемите, които не са част от процесорното ядро, се наричат *периферии*;
- Поради широката употреба на някои от тях съществува варианта за вграждане вътре в процесорния чип;
- Въпреки това те не се превръщат в непериферии вътре, ами си остават вън от ядрото;
- Комуникацията с него се усещствава чрез прекъсвания и I/O регистри;
- Според това разделение перифериите могат да се определят като вътрешни & външни - *internal (on-chip) & external*;

Какво е периферия? ii

- Традиционни примери за вътрешни периферии са таймери и серийни портове;
- Външни периферии могат да бъдат всякакви допълнителни устройства, които закачае към микроконтролера ни и с които комуникираме посредством посочените механизми.

Какво е периферия? iii



Какво е таймер? i

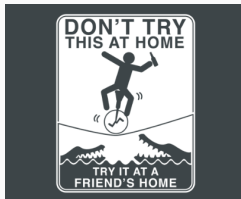
- Таймерите са устройства, чрез които отмерваме интервали от време;
- Обикновено се свързват с периодично извършване на някакво зададено действие;

Пример

- Писукащото нещо на печката, което ни подсказва да не изгорим манджата
- `cron`

Какво е таймер? ii

```
void on_repeat_forever(int numseconds, void (*fun)()) {  
    for (;;) {  
        for (int i = 0; i < numseconds * C; ++i)  
            ;  
        (*fun)();  
    }  
}
```

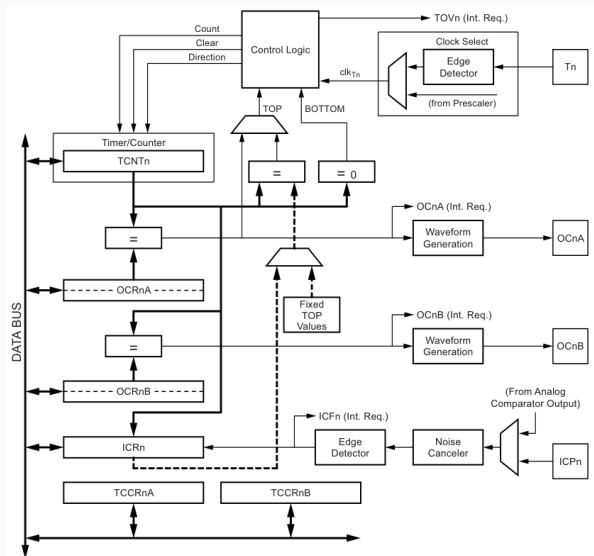


Какво е таймер? iii

- Реализацията е изцяло в хардуер;
- Понеже е периферия, то ресурсите на процесора не биват използвани;
- По този начин можем да извършваме няколко задачи едновременно.

Таймери в ATmega328P

- Има три - Timer0..2;
- Съответно описани в глави 14, 15 и 17;
- Два от тях (Timer0 & Timer2) са 8-битови, докато Timer1 е 16-битов;
- Ние ще използваме предимно Timer1.



Относно Timer1 i

Някои от неговите функционалности са:

- Два отделни сравняващи елемента;
- Auto-reload;
- 16-битов PWM.

Използваме два основни режима

- Изходен - генерират се правоъгълни импулси с конфигурируеми честота, коефициент на запълване и поляритет;
- Входен - най-често засича интервали от време.

Относно Timer1 ii

Основните регистри, които използваме за конфигурация са:

- TCNT1 - timer counter (16b);
- OCR1A/B - output compare register (16b);
- ICR1 - input capture register (16b);
- TCCR1A/B/C - control registers A, B & C (8b);
- TIFR1 - interrupt flag register (8b);
- TIMSK1 - interrupt mask register (8b);

Относно Timer1 iii

Бележка

Понеже ATmega328P е 8-битов процесор, имаме усложнения, когато достъпваме 16-битови регистри. Използва се допълнителен временен регистър, което ни принуждава да се погрижим за атомарността на операцията - т.е трябва да спрем прекъсванията преди да достъпваме 16-битови регистри.

Относно Timer1 iv

```
TCNT1 = 0xaabb;
```

```
ldi r17, 0xaa
```

```
ldi r16, 0xbb
```

```
out TCNT1H, r17
```

```
out TCNT1L, r16
```

the counter, and counter low (TCNT1L) containing the lower eight bits. The TCNT1H register can only be indirectly accessed by the CPU. When the CPU does an access to the TCNT1H I/O location, the CPU accesses the high byte temporary register (TEMP). The temporary register is updated with the TCNT1H value when the TCNT1L is read, and TCNT1H is updated with the temporary register value when TCNT1L is written. This allows the CPU to read or write the entire 16-bit counter value within one clock cycle via the 8-bit data bus. It is important to notice that there are special cases of writing to the TCNT1

[Секция 15.5, стр. 95](#)

Относно Timer1 v

Бележка

Най-голямата целочислена стойност, която се побира в 16-битов регистър е $0xffff = 2^{16} - 1 = 65535$.

Относно Timer1 vi

- Стойността на OCR1A/B се сравнява със стойността на TCNT1;
- Изходът от това сравнение може да бъде използвано за ШИМ върху изводите с маркер OC1A/B (вж. pinout);
- Или пък да се задейства прекъсване;
- Това зависи от конфигурацията на таймера;

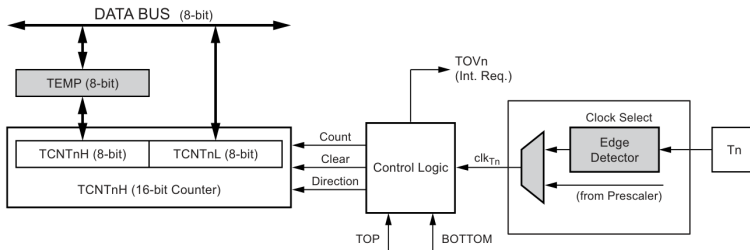
Timer1 - метод на действие i

Брояч (Counter Unit)

- Най-общо казано стойността на TCNT1 се повишава, започвайки от BOTTOM, докато не достигне TOP, където TOP със сигурност е не повече от 0xffff, която стойност наричаме MAX;
- Това се извършва при всеки тактов сигнал;
- Ако такъв липсва, то таймерът е спрян.

Timer1 - метод на действие ii

Figure 15-2. Counter Unit Block Diagram



Signal description (internal signals):

Count	Increment or decrement TCNT1 by 1.
Direction	Select between increment and decrement.
Clear	Clear TCNT1 (set all bits to zero).
clk_{T1}	Timer/Counter clock.
TOP	Signalize that TCNT1 has reached maximum value.
BOTTOM	Signalize that TCNT1 has reached minimum value (zero).

[Секция 15.5, стр. 95](#)

Timer1 - метод на действие iii

Input Capture Unit

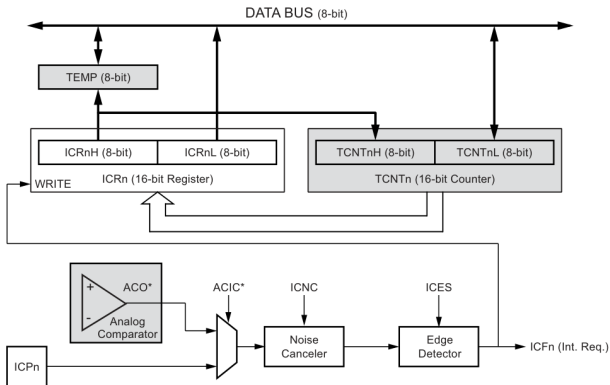
- Таймерът поддържа функционалността да засича външни събития, като запазва текущата стойност на TCNT1 в ICR1 при възникване на такива;
- За целта се следи извода ICP1 и при възникване на преден фронт стойността на брояча се копира, а флагът ICF1 се вдига;
- Също така може да бъде предизвикано и input capture прекъсване, стига то да е конфигурирано;
- Регистърът ICR1 може да има и други употреби, зададени според режима на таймера;

Timer1 - метод на действие iv

- Не се очаква неговата стойност да бъде променяна софтуерно, с изключение на случаите когато там съхраняваме TOP стойността;
- Можем да задействаме input capture софтуерно, ако контролиране ICP1 извода;
- Трябва да се вземе под внимание фактът, че ако се опитваме да засичаме външни събития по този начин, то е необходимо да си подсигурим, че разполагаме с достатъчно процесорно време, за да прочитаме запазените стойност на ICR1, тъй като е възможно да се презапишат.

Timer1 - метод на действие v

Figure 15-3. Input Capture Unit Block Diagram



[Секция 15.5, стр. 96](#)

Timer1 - метод на действие vi

Output Compare Unit

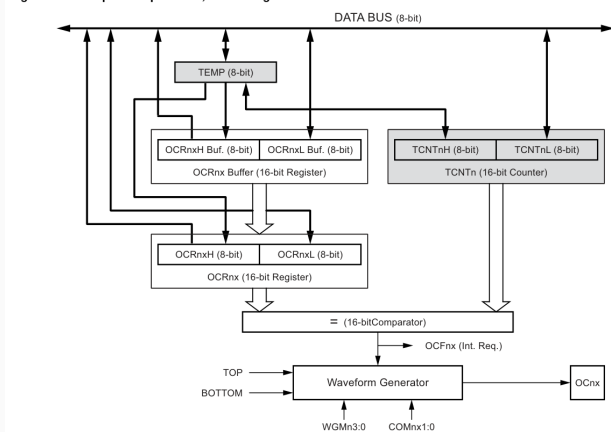
- На всеки такт стойността на брояча се сравнява с тази на OCR1x регистрите;
- Ако се изравнят, то на следващия такт бива вдигнат флага OCF1x, което предизвиква прекъсване (т.нар *compare-match*), когато такова е конфигурирано в mask регистъра;
- Независимо от наличието на прекъсване, “сигналът” за изравняване (match) се използва, за да се създаде изходен сигнал според зададения режим на таймера;

Timer1 - метод на действие vii

- Стойностите на OCR1x се буферират при употребата на ШИМ режимите, а в останалите (СТС & нормалня режим) буферът се заобикаля - по този начин се предотврътява наличие на несиметричните импулси;

Timer1 - метод на действие viii

Figure 15-4. Output Compare Unit, Block Diagram



[Секция 15.5, стр. 98](#)

Timer1 - метод на действие ix

Compare Match Output Unit

- Използва се предимно за определяне на състоянието на изхода (инвертирано или не);
- Извод, който се използва в работещ compare match output unit, не може да бъде употребяват като GPIO;
- Въпреки DDR задава посоката му, която трябва да бъде конфигурирана като изходна.

Timer1 - метод на действие x

Режим на действие

- Режимът на действие наричаме поведението на таймера и неговите изходни изводи (output compare pins);
- Определя се от два *“по-малки”* режима - този на генериране на вълни (waveform generation mode, WGM) и този на изходните пинове (compare output mode, COM);
- При ШИМ режимите COM битовете определят дали сигналът е инвертиран или не; при останалите - какво е поведението на изхода при изравняване (compare match);
- Режимите попадат в няколко категории - нормален, СТС и ШИМ;

Timer1 - метод на действие xi

Нормален

- Най-простият режим на работа;
- Броячът започва от BOTTOM и продължава до MAX без възможност за изчистване при друга стойност;
- Когато задмине MAX, т.е прелее до BOTTOM, флагът TOV1 бива вдигнат;
- Употребата на input capture, както и на двата output compare units е без особени случаи.

Timer1 - метод на действие xii

СТС

- СТС е акроним за clean timer on compare;
- Използваме някой от регистрите OCR1A/ICR1, за да посочим стойност за TOP;
- Подобно на предишния случай (Нормален режим) - флагът за преливане се вдига при достигане на BOTTOM;
- СТС режимите могат да генерират сигнали на изходните изводи, но честотата е значително по-ниска от тази на режимите в бърза ШИМ;

Timer1 - метод на действие xiii

ШИМ

- Броячът се изменя в интервала [BOTTOM; TOP], като прелива в BOTTOM;
- При неинвертиращ режим (зависи от COMx битовете) изходният извод се вдига при изравняване на OCR1x с TCNT1, докато е свален преди това;
- Има възможност да достигне два пъти по-висока честота от тази на CTC режимите;
- “Стандартните” флагове се променят сходно на останалите режими.

Конфигурация на Timer1 i

Трябва да изпълним стъпките

- Задаване на режим;
- Избор на TOP стойност;
- Посочване на делител;
- Избор на инвертираност;
- Обработка на изключение;

Използвайки (най-вече) регистрите TCCR1A & TCCR1B.

Конфигурация на Timer1 ii

Честота

- Нека честота, към която се стремим за изходния извод OC1x е f ;
- Трябва да изберем такава стойност за TOP, така че да е удовлетворено равенството

$$TOP = \frac{f_{clkIO}}{N \cdot f} - 1$$

- Искаме да минимизираме стойността за N в допустимите стойности за делител (prescaler), така че TOP да се побира в 16-битов регистър.

Пример i

Да се конфигурира канал В на таймер 1 в режим на бърза широчинна-импулсна модулация. Режимът на изходния пин да бъде неинвертиращ. Нека честотата бъде 5kHz, а коефициентът на запълване се контролира от свързан DIP ключ - напр. при избрана стойност 5 и максимална 15, то за коефициент на запълване се вземе 33.3%.

Пример ii

Да се конфигурира канал В на таймер 1 в режим на бърза широчинна-импулсна модулация. Режимът на изходния пин да бъде неинвертиращ. Нека честотата бъде 2kHz и се изпълнява прекъсване при преливане.

Литература

- **"AVR Instruction Set Manual"**. URL: <https://ww1.microchip.com/downloads/en/devicedoc/atmel-0856-avr-instruction-set-manual.pdf> (дата на посещ. 08.11.2023)
- **"Atmel328P Datasheets"**. URL: https://gitlab.com/tues-embedded/vmks/-/blob/master/DatasheetsAtmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf?ref_type=heads/ (дата на посещ. 28.09.2023)

- V. Garistov. **"Configuring ATmega328P Timers"**. URL:
https://gitlab.com/tues-embedded/vmks/-/blob/master/Other%20Useful%20Files/%D0%A0%D1%8A%D0%BA%D0%BE%D0%B2%D0%BE%D0%B4%D1%81%D1%82%D0%B2%D0%BE%20%D0%B7%D0%B0%20%D0%BA%D0%BE%D0%BD%D1%84%D0%B8%D0%B3%D1%83%D1%80%D0%B8%D1%80%D0%B0%D0%BD%D0%B5%20%D0%BD%D0%B0%20%D1%82%D0%B0%D0%B9%D0%BC%D0%B5%D1%80%20%D0%B2%20%D1%80%D0%B5%D0%B6%D0%B8%D0%BC%20%D0%BD%D0%B0%20%D0%B1%D1%8A%D1%80%D0%B7%D0%B0%20%D0%A8%D0%98%D0%9C%20%D0%B2%20ATmega328.pdf?ref_type=heads (дата на посещ. 08.11.2023)